

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Applicant: COHEN	§	
	§	
Serial No.: 10/826,503	§	
	§	
Filed: April 19, 2004	§	Group Art Unit: 2134
	§	
For: METHOD FOR PREVENTING	§	Attorney
ACTIVATION OF	§	Docket: 2808/28
MALICIOUS OBJECTS	§	
	§	
Examiner: Jacob Lipman	§	

Commissioner of Patents and Trademarks
Washington, D.C. 20231
ATTENTION: Board of Patent Appeals and Interferences

APPELLANT'S BRIEF

Dear Sir:

This is in furtherance of the Notice of Appeal filed in this case on October 28, 2007.

The fees required under § 1.17(f) and any required petition for extension of time for filing this brief and fees therefor are dealt with in the accompanying TRANSMITTAL OF APPEAL BRIEF.

This brief is transmitted in triplicate.

This brief contains these items under the following headings and in the order set forth below:

- I. REAL PARTY IN INTEREST
- II. RELATED APPEALS AND INTERFERENCES
- III. STATUS OF CLAIMS
- IV. STATUS OF AMENDMENTS
- V. SUMMARY OF CLAIMED SUBJECT MATTER

- VI. GROUNDS OF REJECTION TO BE REVIEWED ON APPEAL
- VII. ARGUMENTS - REJECTION UNDER 35 USC §103(a)
- VIII. APPENDIX OF CLAIMS INVOLVED IN THE APPEAL
- IX. APPENDIX OF EVIDENCE
- X. APPENDIX OF RELATED PROCEEDINGS

I. REAL PARTY IN INTEREST

Aladdin Knowledge Systems, Inc, is the real party in interest, and is the assignee of Application No. 10/826,503.

II. RELATED APPEALS AND INTERFERENCES

The Appellant's legal representative, or assignee, does not know of any other appeal or interferences which will affect or be directly affected by or have bearing on the Board's decision in the pending appeal.

III. STATUS OF CLAIMS

The subject patent application was originally filed with 11 claims. During the course of prosecution, Claim 3 was cancelled. Claims 1, 2, and 4-11 are presently pending in the application, and all stand rejected.

A Notice of Appeal was filed on September 4, 2007, appealing the Office Action mailed April 9, 2007, finally rejecting Claims 1, 2, and 4-11.

A. TOTAL NUMBER OF CLAIMS IN APPLICATION

Claims in the application are: 1, 2, and 4-11

B. STATUS OF ALL THE CLAIMS

1. Claims cancelled: 3
2. Claims withdrawn from consideration but not cancelled: none
3. Claims pending: 1, 2, and 4-11
4. Claims allowed: none
5. Claims rejected: 1, 2, and 4-11

C. CLAIMS ON APPEAL

The claims on appeal are: 1, 2, and 4-11

IV. STATUS OF AMENDMENTS

One amendment was filed in response to the Final Office Action mailed April 9, 2007, canceling claim 3. In the Advisory Action Before Appeal mailed June 13, 2007, this proposed amendment was entered for purposes of appeal.

V. SUMMARY OF CLAIMED SUBJECT MATTER

The subject matter of the presently-pending claims is directed to a method for protecting a computer from attack by malicious executable data objects that may be sent to that computer via a data communications network, in such a way that minimizes the transmission delay in delivering data objects to the computer.

At a checkpoint through which data objects pass on their way to the computer, a data object is enclosed within an envelope file created at the checkpoint to contain that specific data object — it is this envelope file containing the data object, rather than the data object itself, which is thereafter sent to the computer. The envelope file is executable, having code for extracting the data object contained therein, and also has an integrity indicator to show whether the contained data object is benign or malicious.

During the time period over which an envelope file is transmitted from the checkpoint to the computer over the network (one data packet at a time), the data object that was placed in the envelope is inspected at the checkpoint to determine if that data object is benign or malicious. If the data object is benign, the integrity indicator of the envelope file containing the data object is marked as “benign”. Then the transmission to the computer is completed, whereupon the data object may be extracted from the envelope file by using the extraction code therein. If, however, the data object is malicious, then the integrity indicator is marked as “malicious” so that other appropriate action may be taken — such as deleting the data object, alerting the user to the malicious content, and so forth.

In addition to the use of an envelope file to enclose and isolate potentially malicious data objects, at least one part of the envelope file is withheld at the checkpoint until the inspection process is complete and a determination of the

contained data object is made. In this manner, the bulk of the envelope file containing the data object can be sent to the computer without waiting for the inspection to complete, thereby minimizing the transmission delay. At the same time, however, withholding a part of the envelope file guarantees that in the case of a malicious contained data object, no portion of the malicious data object can be activated at the computer, because a complete envelope file is necessary before extraction of the contained data object can be performed. The transmission delay is minimal because only the withheld part of the envelope file need be transmitted to the computer upon completion of the inspection, and this part may be as small as a single data packet.

In the above manner, the present invention provides a method for preventing the activation of malicious data objects at the computer while minimizing transmission delays.

The key features above are recited as limitations in independent claim 1 of the present application, and all remaining presently-pending claims depend from claim 1.

Independent claim 1 recites the following, with support in the specification as indicated in parentheses by paragraph numbers and drawing reference numbers as appearing in the published application (United States Patent Application Publication number **US 2005/0235160 A1**):

1. A method for preventing activating a malicious object (paragraphs [0007], [0008]) passing through a checkpoint, and decreasing the overall inspection delay thereof (paragraphs [0009], [0011], [0022] and [0023]), the method comprising the steps of:
 - a. at said checkpoint (paragraph [0002], Figure 1 number 30), creating an envelope file (Figure 3 number 102), being an executable file (paragraph [0024], Figure 2 number 50, paragraph [0011]) comprising:

- said object (paragraphs [0011], [0025], Figure 2 number 52);
- code (Figure 2 number 51) for extracting said object from said envelope file (paragraphs [0011] and [0027]); and
- an indicator (Figure 2 number 53) for indicating the integrity of said object (paragraph [0026]);
- b. forwarding said envelope file instead of said object toward its destination (Figure 3 numbers 101, 105, 106, 107, and 108), while holding at least a part of said envelope file which comprises said indicator (paragraph [0043]);
 - c. inspecting said object (Figure 2 number 105, paragraph [0040]);
 - d. setting said indicator on said envelope file to indicate the inspection result thereof (paragraphs [0011], [0030]), and
 - e. releasing the rest of said envelope file (paragraph [0011], Figure 3 number 108).

The above-cited paragraphs of the present specification, or relevant excerpts thereof, appear on the noted page numbers and read as follows:

page 1 [0002] ... the gateway to a local network is a proper point for checking out objects (e.g. files and email messages) that pass through it...

page 1 [0007] It is an object of the present invention to provide a method for preventing activation of malicious objects.

page 1 [0008] It is a further object of the present invention to provide a method for preventing from a checkpoint the activation of malicious objects on the executing platform.

page 1 [0009] It is a still further object of the present invention to provide a method for inspecting a file on a checkpoint, by which the delay thereof is decreased in comparable to the prior art.

page 1 [0011] A method for preventing activating a malicious object passing through a checkpoint, and decreasing the overall inspection delay thereof, the method comprising the steps of: (a) at the checkpoint, creating an envelope file, being an executable file comprising: the object; code for extracting the object from the envelope file; and an indicator for indicating the integrity of the object; (b) forwarding the envelope file instead of the object toward its destination, while holding at least a part of the envelope file which comprises the indicator; (c) inspecting the object; and (d) setting the indicator on the envelope file to indicate the inspection result thereof, and releasing the rest of the envelope file.

page 2 [0022] ... a gateway server that inspects data that is transferred through it ... has to hold an inspected object until its harmlessness is indicated, on the other hand holding the file at the gateway may cause a bottleneck to the data traffic passing through the gateway.

page 2 [0023] ... The present invention allows passing a file toward its destination while inspecting the file, and still preventing the activation of the file at the destination site until its integrity is indicated. ... instead of forwarding the original file to its destination, a substitute file which comprises the original file is constructed and transmitted to the destination. The substitute file is also referred herein as envelope file.

page 2 [0024] ... the envelope file is an executable which comprises the following parts:

page 2 [0025] the original file {the "object", per paragraph [0002]};

page 2 [0026] an indicator about the integrity of the original file;
and

page 2 [0027] an executable part, upon which execution extracts the
original file, and executes it.

page 2 [0030] For example, in the case where the integrity of the
original file is indicated at the gateway, the gateway sends the true
CRC value of the original file. In case the original file is indicated as
comprising malicious content, the gateway sends a wrong CRC
value of the original file. As well, other indicators can be used.

page 2 [0040] At block 105, a copy of the packet is sent to the
inspection facility for inspection.

page 2 [0043] ... according to a preferred embodiment of the
invention, the indicator is stored within the last part of the envelope
file. This way usually (but not always) the integrity indicator will be
the last to reach the destination.

VI. GROUNDS OF REJECTION TO BE REVIEWED ON APPEAL

In a final Office Action mailed April 9, 2007 (herein “the Office Action”), the Examiner rejected the presently-pending claims under 35 USC §103(a) as being unpatentable over US Patent Number 6,088,803 to Tso, et al., in view of US Patent Publication Number 2004/0054928 of Hall.

Claims under appeal include independent claim 1; and dependent claims 2 and 4 through 11, all of which depend from claim 1.

VII. ARGUMENTS - REJECTION UNDER 35 USC §103(a)

In a Response to the Office Action, filed June 6, 2007 (herein “the Response”, which is hereby incorporated by reference into the present Appellant’s Brief), the Applicant respectfully traversed the Office Action’s 35 USC §103(a) rejection on the grounds that, to a person having ordinary skill in the art, it is technically not possible to combine Tso and Hall in a manner that achieves the structural limitations recited in presently-pending claim 1. Therefore, there is no reasonable expectation of success in combining Tso and Hall, and thus such a combination does not meet the criteria of obviousness.

Tso discloses a system for inspecting a data object that is being sent to a computer over a data network, in which a part of the data object is withheld pending the completion of the inspection process.

Tso, however, fails to disclose an envelope file containing the data object. The envelope file and the components thereof are recited as limitations in presently-pending independent claim 1. The present invention teaches that the envelope file is necessary to prevent activation of a partially-received file (such as a script), as disclosed in the specification of the present application:

page 1 [0005] ... This solution {such as taught by Tso} is applicable only for files that in order to be executed or activated, the whole file has to be available on the executing platform. However, if the executing platform activates a file even in the case where only a part of the file is available, the executing platform is exposed to viruses and other malicious forms.

Because Tso fails to disclose an envelope file, Tso also fails to disclose executable code for extracting the data object from an envelope file. Moreover, Tso fails to disclose an integrity indicator, which is additionally part of the envelope file.

Because this executable code and the integrity indicator are limitations recited in presently-pending claim 1, Tso thus fails to meet all the limitations of the present claims.

The Examiner, however, has construed Tso's data object itself to be an envelope file ("a copy of the object missing the final segment", Final Office Action mailed April 9, 2007, page 3 line 11). The Appellant respectfully traverses this interpretation, because the specification of the present application clearly defines an envelope file for a data object (*inter alia* in Figure 2) as a file **50** containing the data object **52** and also containing additional elements **51** and **53**, and therefore being materially more than just an incomplete copy of the data object itself. The elements of this definition are included as limitations of the pending independent claim 1.

Despite construing Tso's data object without the withheld part as an envelope file, however, the Examiner does take note of the fact that Tso fails to disclose the inclusion of executable code which extracts the object (Office Action page 3 lines 11-12).

The Examiner cites Hall as using an "executable wrapper" to protect files, stating that "Hall discloses an executable wrapper used to protect files", and that "Hall teaches that the wrapper protects a file from being executed without checking an authorization algorithm by hiding the object, and supplementing an executable wrapper that will release the object if authorization is checked" (Office Action page 3 lines 12-16, Hall paragraphs [0044]-[0045]). This is the basis for the current 35 USC §103(a) rejection of the pending claims as obvious over Tso in view of Hall.

The Appellant respectfully traverses the §103(a) rejection, as detailed below.

Hall fails to disclose any material related to the structural limitations of the presently-pending claims or capable of being combined with Tso or to modify Tso to

achieve those limitations. For example, Hall fails to teach or reasonably suggest executable code that extracts a data object from another file, such as an envelope file.

Hall is directed to a method for preventing unauthorized personnel from executing fully-installed software (such as routines implementing operating system commands) within a computer system. Hall teaches relocating those system command routines from their standard locations to hidden locations, and substituting proxy executables (which Hall refers to as “wrappers”) in place thereof in the standard locations. These “wrappers” (proxy executables) have the same name as the original system commands, but contain only code for checking user authorization and conditionally executing the relocated original system command software. Thus, when a user tries to execute one of the protected system commands, he or she will instead be executing the “wrapper” (a proxy). The “wrapper” proxy then checks the user’s authorization status, and only if that user is permitted to access the desired system command is that system command actually executed by the “wrapper” proxy. It is emphasized that Hall fails to teach or reasonably suggest that a “wrapper” contains or comprises the system command software itself. Hall also fails to teach or reasonably suggest executable code for extracting a data object from a file containing that data object.

Combining Tso with Hall does not teach or reasonably suggest all the limitations of presently-pending independent claim 1. Hall’s “wrappers” are structurally completely different from the envelope files of the present claims. A “wrapper” according to Hall does not contain or comprise another data object, as does the envelope file of the present claims. Moreover, as noted, a “wrapper” according to Hall does not contain or comprise code for extracting any data object, nor an integrity indicator for any other data object.

Tso and Hall, individually as well as combined, fail to teach or reasonably suggest putting a data object passing through a checkpoint into an executable envelope file having code for extracting that data object, as recited in presently-pending independent claim 1. Moreover, the combination of Tso and Hall does not have a reasonable expectation of success in meeting the limitations recited in the presently-pending claims.

In an Advisory Action mailed June 13, 2007 (herein "the Advisory Action"), the Examiner claimed that "Applicant argues that combining Tso and Hall would not have been obvious to one of ordinary skill in the art since Tso is protecting from malicious code to be installed, and Hall is protected from unauthorized execution." The Appellant respectfully notes that this characterization of the Applicant's argument in the Response is incomplete. As clearly put forth in the Response, a person of ordinary skill in the art would realize that it is technically *not possible* to combine Tso and Hall in a manner that meets the limitations of the presently-pending claims; and therefore Tso in view of Hall fails to teach or reasonably suggest all the claim limitations; and further that Tso in view of Hall as proposed in the Office Action fails to have a reasonable expectation of success. Meeting all claim limitations and a reasonable expectation of success are both stipulated, *inter alia* in MPEP 2143, as necessary for a *prima facie* case of obviousness under 35 USC §103(a). The Appellant respectfully maintains that, absent these two necessary elements, Tso in view of Hall as proposed in the Office Action fails to sustain a 35 USC §103(a) rejection.

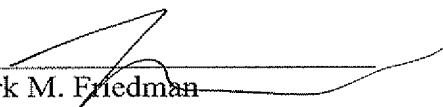
The Appellant further notes that in both the Office Action and the Advisory Action, the Examiner acknowledged that Tso and Hall protect against different types of attacks against the computer, and proposed that there is a motivation to combine

Tso and Hall in order to protect against different types of attack. The Appellant respectfully maintains that this suggested use of those two methods to protect against two different types of attack does not constitute a bona fide combination of the methods, nor a modification of one method by the other. Such a suggestion would at most amount to a sequential application of two different and independent methods for two different and independent purposes that have no structural interrelationship. That is not at all the same as *combining* those methods, or *modifying* one method with the other.

Tso and Hall are wholly independent of one another and *cannot be combined*; moreover, Tso *cannot be modified* by Hall to achieve the structural limitations of the present independent claim 1, nor is there any reasonable expectation of success in attempting to combine them or to use one to modify the other to achieve those structural limitations of claim 1.

No matter how Tso and Hall are used, neither of them alone nor both of them together provide creating an envelope file containing the original object and executable code for extracting the original object, as recited in pending independent claim 1.

Respectfully submitted,



Mark M. Friedman
Attorney for Applicant
Registration No. 33,883

Date: December 9, 2007

VIII. APPENDIX OF CLAIMS INVOLVED IN THE APPEAL

The text of the claims on appeal is:

1. A method for preventing activating a malicious object passing through a checkpoint, and decreasing the overall inspection delay thereof, the method comprising the steps of:

- a. at said checkpoint, creating an envelope file, being an executable file comprising: said object; code for extracting said object from said envelope file; and an indicator for indicating the integrity of said object;
- b. forwarding said envelope file instead of said object toward its destination, while holding at least a part of said envelope file which comprises said indicator;
- c. inspecting said object;
- d. setting said indicator on said envelope file to indicate the inspection result thereof, and
- e. releasing the rest of said envelope file.

2. A method according to claim 1, wherein said checkpoint is selected from a group comprising: a gateway server, a proxy server.

3. (Canceled)

4. A method according to claim 1, wherein the name of said envelope file is identical to the name of the inspected object.

5. A method according to claim 1, wherein the name of said envelope file differs than the name of the inspected object.

6. A method according to claim 1, wherein said indicator is selected from a group comprising: a CRC of at least one part of said envelope file, a CRC of at least one part of said inspected object, a checksum of at least one part of said envelope file,

a checksum of at least one part of said inspected object, a value stored within said envelope file, absence of a part of said envelope file, absence of a part of said object.

7. A method according to claim 1, wherein at least a part of said object is secured.

8. A method according to claim 1, wherein at least a part of said envelope file is secured.

9. A method according to claim 1, wherein said indicator is stored within the last part of said envelope file.

10. A method according to claim 1, wherein said envelope file further comprises code for displaying an acknowledgment.

11. A method according to claim 10, wherein said acknowledgment indicates integrity of said object.

IX. APPENDIX OF EVIDENCE

NONE

X. APPENDIX OF RELATED PROCEEDINGS

NONE